

IN THE CLAIMS:

1. (Currently Amended): A method for exchanging objects between two computing entities in an object-oriented programming environment using a transport mechanism in which said data units objects are contained in files, each file defining a resource, each resource designed to contain a plurality of particular ones of said objects, said method comprising the steps of:

- (1) providing a resource factory for building resources, said factory including a plurality of software modules for building resources from a data source, each said software module designed to build a resource of a particular type;
- (2) responsive to a request for an object from a first computing entity, selecting a software module for building a resource of the type to which said object corresponds;
- (3) building a resource for containing said object using said selected software module, said resource populated with information defining said resource, but not containing said object;
- (4) inserting said object into said resource;
- (5) transmitting said resource to said first computing entity using said transport mechanism; and
- (6) providing to said first computing entity said object.

2. (Previously Presented): The method of claim 1 wherein, in step (4), only said object is inserted in said resource.

3. (Previously Presented): The method of claim 2 further comprising the steps of:

(7) providing a reflection adapter factory for populating objects within resources, said factory adapted to provide software modules for populating objects, each said software module designed for an environment corresponding to an object;

(8) responsive to a request for a property of said object, selecting a one of said reflection adapters for the environment of the particular property;

(9) populating said object with said property; and

(10) providing to said first computing unit said property.

4. (Original): The method of claim 3 wherein said object comprises a plurality of properties and step (9) comprises populating said object with all properties of said object that can be reflected.

5. (Currently Amended): A method for exchanging objects between two computing entities in an object-oriented programming environment using a transport mechanism in which said ~~data units~~ objects are contained in files, each file defining a resource, each resource designed to contain a plurality of particular ones of said objects, said method comprising the steps of:

(1) providing a resource factory for building resources, said factory including a plurality of software modules for building resources from a data source, each said software module designed to build a resource of a particular type;

(2) determining whether said first computing entity has stored a resource containing said object;

(3) if said first computing entity has stored a resource corresponding to said object, determining if said corresponding resource stored at said first computing entity contains said object;

(4) if said corresponding resource stored at said first computing entity does not contain said object, said first computing entity issuing a request for said object ;

(5) responsive to a request for said object from said first computing entity, selecting a software module for building a resource of the type to which said object corresponds;

(6) building a resource for containing said object using said selected software module, said resource populated with information defining said resource, but not containing said object;

(7) inserting only said object into said resource;

(8) transmitting said resource to said first computing entity using said transport mechanism; and

(9) providing to said first computing entity said object.

6. (Currently Amended): A method for exchanging objects between two computing entities in an object-oriented programming environment using a transport mechanism in which said ~~data units~~ objects are contained in files, each file defining a resource, each resource designed to contain a plurality of particular ones of said objects, said method comprising the steps of:

(1) providing a resource factory for building resources, said factory including a plurality of software modules for building resources from a data source, each said software module designed to build a resource of a particular type;

- (2) responsive to a request for an object from a first computing entity, selecting a software module for building a resource of the type to which said object corresponds;
- (3) building a resource for containing said object using said selected software module, said resource populated with information defining said resource, but not containing said object;
- (4) inserting said object into said resource;
- (5) transmitting said resource to said first computing entity using said transport mechanism;
- (6) providing to said first computing entity said object.
- (7) providing a reflection adapter factory for populating objects within resources, said factory adapted to provide software modules for populating objects, each said software module designed for an environment corresponding to an object;
- (8) determining whether said first computing entity has stored said property;
- (9) if said first computing entity has not stored said property, issuing a request for said property;
- (10) responsive to said request for said property of said object, selecting a one of said reflection adapters for the environment of the particular property;
- (11) populating said object with said property; and
- (12) providing to said first computing unit said property.

7. (Previously Presented): The method of claim 3 wherein said transport mechanism comprises XML document.

8. (Original): The method of claim 7 wherein said objects comprise Java objects.

9. (Previously Presented): The method of claim 8 wherein said transport mechanism comprises an XMI document.

10. (Original): The method of claim 9 wherein steps (4) and (5) utilize the Meta Object Facility of the Object Management Group specification to read an XMI document.

11. (Original): The method of claim 8 wherein, in step (2), said information defining said resource comprises at least a package object of said resource.

12. (Currently Amended): A method for exchanging objects between two computing entities in an object-oriented programming environment using a transport mechanism in which said ~~data units~~ objects are contained in files, each file defining a resource, each resource designed to contain a plurality of particular ones of said objects, said method comprising the steps of:

(1) providing a resource factory for building resources, said factory including a plurality of software modules for building resources from a data source, each said software module designed to build a resource of a particular type;

(2) responsive to a request for an object from a first computing entity, selecting a software module for building a resource of the type to which said object corresponds;

(3) building a resource for containing said object using said selected software module, said resource populated with information defining said resource, but not containing said object;

(4) inserting said object into said resource;

(5) transmitting said resource to said first computing entity using said transport mechanism;

(6) providing to said first computing entity said object.

(7) providing a reflection adapter factory for populating objects within resources, said factory adapted to provide software modules for populating objects, each said software module designed for an environment corresponding to an object;

(8) determining whether said first computing entity has stored said property;

(9) if said first computing entity has not stored said property, issuing a request for said property;

(10) responsive to a request for said property of said object, selecting a one of said reflection adapters for the environment of the particular property;

(11) determining whether said selected reflection adapter has previously reflected said requested property;

(12) if said first computing entity has previously reflected said requested property, populating said object with said property; and

(13) providing to said first computing unit said property.

13. (Original): The method of claim 1 wherein said data source for building said resources comprises a live system.

14. (Original): The method of claim 1 wherein said data source for building said resources comprises a database.

15. (Original): The method of claim 1 wherein said data source for building said resources comprises a document in a format other than a format of said transport mechanism.